RICE Repository Process

8 July 2003

1 INTRODUCTION

1.1 What is the RICE Repository?

During a COTS implementation there are additional configuration, design, and/or programming requirements necessary to satisfy functional requirements and achieve the desired functionality. RICE objects represent requirements not currently supported within the core functionality of the COTS product being implemented and therefore require additional technical development. This development enables: the creation of unique Reports not standard in the product, the creation of Interfaces to external systems, the creation of Conversion programs to transfer data from an obsolete system to the new system and the creation of Enhancements (or Extensions) allowing additional functionality to be added to the system without disturbing the core software code. RICE the acronym that stands for Reports, Interfaces, Conversions and Extensions are further defined as follows to ensure consistency across programs and within the RICE Repository:

Report - A formatted and organized presentation of data.

Interface - A boundary across which two independent systems meet and act on or communicate with each other.

Conversion - A process that transfers or copies data from an existing system to load production systems.

Extension - A program that is in addition to an exiting standard program but that does not change core code or objects.

The RICE repository is the collection of specific elements related to each RICE object. These data elements may include object description and attributes, reference documentation, point of contact information and development status.

1.2 Where is the RICE Repository?

The repository currently resides in a series of Excel spreadsheets and can be accessed via the EI Toolkit at http://deskbook.dau.mil click ‘Software Tools’ within the ‘Program Management’ category or by logging onto QuickPlace.

1.3 Purpose of the RICE Repository

The purpose of the RICE Repository is to create a vehicle in which COTS Implementations can leverage work already done, reduce redundancy, reduce effort and costs. The objectives of the RICE repository are to:

- Provide a central repository to facilitate the collection and sharing of RICE objects and lessons learned

- Provide a vehicle that allows the end user to search, submit, and leverage already developed RICE Objects
- Integrate the sharing and leveraging of RICE Objects into the Acquisition/Contracting, Project Management and Implementation processes
- Realize development cost savings by reusing existing RICE objects or establishing a baseline from an existing object

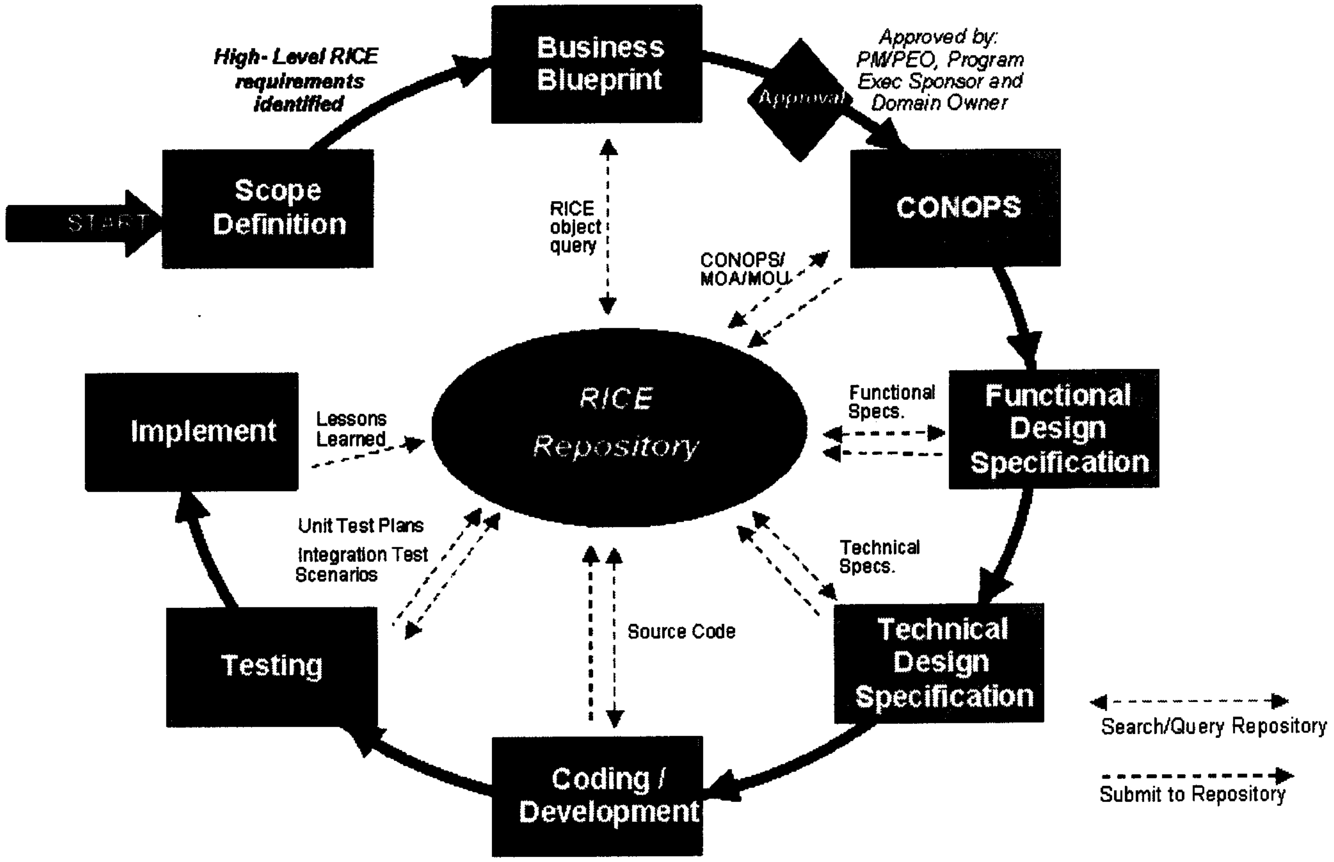
1.4 Timing of Use

Critical to realizing the benefits of the RICE repository is the timing of its use. The repository must be utilized, in most cases, during the Blueprint phase and during the Design & Build phase. The RICE Process must also be included as part of the Systems Integrator RFP/RFQ/SOO and associated contracts thus making the Integrator accountable and responsible for the use and population of the RICE Repository. Refer to the EI Toolkit Acquisition/Systems Integration/RFP/SOO and Contract templates for sample language.

2 RICE REPOSITORY IN THE PROJECT LIFECYCLE

2.1 RICE Object Development Lifecycle

RICE object development is an extensive process that starts with the conceptual design of the software functionality. The initial development of the object requires analysis from the implementation team as well as the owners of the requirements, data, and legacy systems. The RICE object has a lifecycle throughout the project that includes Approval to Develop, Concept of Operations, Functional Design Specification, Technical Design Specification, Coding, Testing and Implementation. The effort that is involved in completing this development cycle can be significant. It is in the best interest of other DoD projects to leverage this investment. The RICE Repository is a source to obtain information, to decrease the required effort, and to ultimately decrease the time and cost of development. Beginning with the Business Blueprint phase the RICE repository is queried for leveragable RICE objects. Subsequently, each phase of the object development cycle requires the repository to be queried for leveragable information and any newly created object information must be submitted to the repository for use by other programs. The diagram below depicts the required use of the RICE Repository in the lifecycle of a COTS implementation.

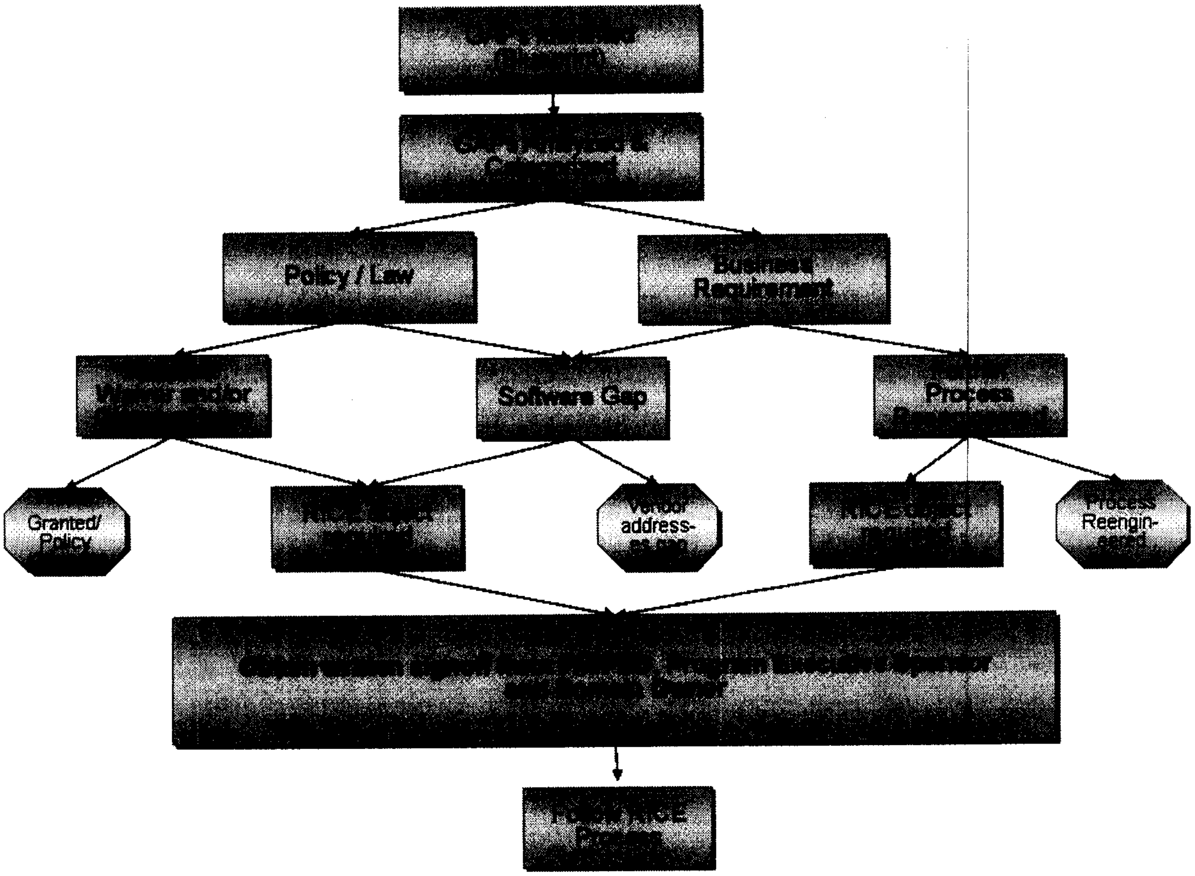


2.1.1 Scope Definition

The project identifies the functionality that will be implemented as part of the project. Desired software functionality is defined and high-level functional requirements are defined. High-level RICE requirements may be identified to plan resource requirements and blueprint activities. Until the RICE objects are further defined, and the project has agreed to proceed on RICE development, there would be limited interaction with the RICE Repository in this phase.

2.1.2 Business Blueprint

The agreed upon scope of the project is reviewed to define the future business processes. Functional requirements are mapped to software functionality and a gap analysis is performed. It is during the Business Blueprint phase that RICE objects will be further defined, analyzed, prioritized and planned. The following diagram represents the analysis and categorization required prior to seeking approval.



2.1.2.1 Gaps Identified

Functionality gaps are identified for further review and action

2.1.2.2 Gaps Analyzed and Categorized

Functionality gaps are reviewed and categorized into two main reason categories: Policy/Law – unique requirements due to a regulatory or statutory requirement and Business Process – requirements due to business processes unique to a certain business area.

2.1.2.2.1 Policy / Law

For gaps resulting from regulatory/statutory requirements a waiver or policy change should be requested by the program. If the waiver is granted then the process should be re-engineered to remove the gap. If a waiver is not received or the policy is not changed add the gap to the list of required RICE objects. The object approval form must state the policy, chapter & paragraph, the policy Domain Owner and whether it was or was not referred to the Domain Owner for assistance.

2.1.2.2.2 Business Process

For gaps resulting from a unique business process requirements further efforts must be undertaken to reengineer the process. If further reengineering does not remove the gap or refer the required functionality to the appropriate software vendor asking for a change to be made to the core software accommodating the requirement. If the process can be reengineered and/or if the software vendor can provide a software change remove the gap from the list. If the vendor is unable to provide a fix the software consider removing the requirement the current scope and moving it to a future release or functionality upgrade or add the gap to the list of required RICE objects. The object approval form must include detail of effort made to reengineer and/or to seek support from the software vendor.

2.1.2.2.3 Obtain Approval from PM/PEO, Program Sponsor & Domain Owner

Any and all RICE objects identified as absolutely required must be approved by the Process Owner, PM, Project Sponsor and the Domain Owner (as part of the Domain Advocacy Approval Process.) This approval must be obtained prior to proceeding to the phase of the RICE Development Process.

2.1.2.2.4 Follow RICE Process

For all objects that are ultimately approved the RICE Process, as further defined in this document, must be followed.

2.1.3 Approval

Refer to Section 2.1.2.2.3 Obtain approval from PM/PEO, Program Executive Sponsor & Domain Owner.

2.1.4 RICE Object CONOPS

When an external party is involved in the development process, an initial object Concept of Operations (CONOPS) and a corresponding Memorandum of Agreement (MOA) or Memorandum of Understanding (MOU) needs to be developed to define the scope, expectations, service agreements, cost, roles and responsibilities, and conceptual design.

2.1.5 Functional Design Specification

The implementation team defines the functional requirements and planned design of the RICE object. Requirements such as the conceptual design, data elements, translation definitions, and formats may be included in the functional specification.

2.1.6 Technical Design Specification

The development team will use the functional design specification and write the technical specifications to satisfy the requirements. The technical specification may include the technical approach, proposed data elements and tables, and proposed coding requirements. Once approved, the technical specification will be used to develop the programming code.

2.1.7 Coding/Development

The development team completes the design/coding of the RICE object based on the specifications. Source code for similar RICE objects can be exchanged via the RICE repository.

2.1.8 Testing

Testing includes any of the testing phases such as: unit, string, integration, and performance/stress (if applicable). The development team as well as the functional team needs to develop testing plans and test cases to execute throughout the testing cycle. The RICE Repository can be used to exchange testing documentation.

2.1.9 Implementation

Once a RICE object is deployed and functioning within the production environment, there are usually inevitable changes or modifications. There are also usually key lessons learned from the development cycle and the implementation that should be documented and made available for others to benefit. These implementation considerations and lessons learned can also be submitted to the RICE Repository for future reference.

2.2 RICE Object Approval Process

Within each COTS implementation project, there must be a defined approval process for each RICE object to be developed. Each object must be justified and authorized prior to development. Only those objects related to previously authorized are to be developed. All objects must have a cover sheet describing its purpose and justification as well as sign-off. To ensure new and modified deliverables are captured and submitted to the repository, the object sign-off sheets must be created/revised to include submission to the RICE repository.

The sign-off sheet must be changed to reflect that the author has verified a similar object/documentation was or was not available in the RICE repository.

Each sign-off sheet will also include the appropriate general information as previously stated (system version, point-of-contact info, etc.) to aid the User querying the repository. The programs will be required to modify their process to gather this information, and send an electronic or scanned copy to the RICE Repository Owner.

Compliance with this process will:

- Add a second authorization point to ensure that an unwarranted or redundant number of development objects are not being created.
- Allow time to revisit BPR for a particular functional process.
- Minimize the need for data calls, by the RICE Sub-Group, in order to obtain updated object data.

2.3 Updating the RICE Repository

2.3.1 Frequency of New Object Data Submittals to the Repository

New or updated RICE object data should be submitted to the RICE Repository Owner according to the RICE Development Cycle and subsequent updates to created objects on at least a quarterly basis. This will keep the repository current and allow users access to all development.

2.3.2 Submitting RICE Object Information to the RICE Repository

Currently, new and updated RICE object information is submitted through the Program Implementation Group (PIG) RICE Sub-Group via data calls to the existing programs. As the Repository technology evolves a more interactive/dynamic mechanisms to submit information will be developed.

2.3.3 RICE Acceptance Process

To ensure completeness and accuracy of the data within the RICE repository, an acceptance process has been enacted. The submissions to the RICE repository will be queued prior to posting. RICE repository resources will have the task of reviewing the submission files, verifying completeness of the submission, contacting the Provider for additional information if necessary, and determining the appropriate placement for the posting.

This acceptance process will ensure the RICE repository will serve the purposes of the Users in an accurate and complete knowledge transfer.

3 APPENDIX A: IMPLEMENTATION PROJECT TEMPLATES

3.1 RICE Object Overview & Approval Template

Section I: Description				
Team / Application Area:			Date (MM/DD/YYYY):	
Requested by:			Phone Number (###-####):	
RICE Object Title:				
Short description:				
Reason for Object Creation:	( ) Policy ( ) Business Process ( ) Software Functionality Gap If Policy is selected please indicate: Waiver/Policy Change Requested: ( ) Yes ( )No Policy Number, Chapter & paragraph: Policy Name: Policy Domain Owner: Referred to Domain Owner: ( ) Yes ( ) No			
Program type:	( ) Conversion ( ) Batch Interface ( ) Online Interface ( ) Report ( ) Form ( ) Enhancement ( ) Extension ( ) Online program ( ) Modification ( ) Unknown ( ) Others			
Priority:	( ) high/mandatory ( ) medium/recommended ( ) low/optional			
Data volume:	_____________ records	Date to be available:		

Section II: Repository Review		
RICE repository queried for similar objects: ( ) Yes ( ) No		
List similar RICE Objects: Name	Ref#	Implementation Project

List related CONOPS and file names:	Percent of reusability	Projected hours saved due to template reusability	Projected hours to consolidate / modify for current use

List related FDS (Functional Design Specs) and file names:	Percent of reusability	Projected hour savings due to template reusability	Projected hours to consolidate / modify for current use
List related TDS (Technical Design Specs) and file names:	Percent of reusability	Projected hour savings due to template reusability	Projected hours to consolidate / modify for current use
List program code names that can be modified and reused (including implementation name):	Percent of reusability	Projected hour savings due to template reusability	Projected hours to consolidate / modify for current use

Section III: Overview

Summarize the applicability of templates from other implementations for this RICE object:

Which documents are most useful from other implementations?

What is the overall cost benefit of having other templates available for use on this implementation?

Section IV: Assumptions

List assumptions that the above information is subject to:

Section V: References

Section VI: Approvals

Business Analyst Signature	Printed Name	Date
Technical Analyst Signature	Printed Name	Date
Business Team Lead Signature	Printed Name	Date
Project Manager/ Project Executive Officer Signature	Printed Name	Date
Project Executive Sponsor Signature	Printed Name	Date
Domain Owner Signature	Printed Name	Date

3.2 Functional Design Specification Template

Section I: Justification			
Area (Module):		Date (20/06/2002):	
Requested by:		Phone Number / Extension (99999):	
Title:			
Short description:			
Program type:	( ) Conversion ( ) Batch Interface ( ) Online Interface ( ) Report ( ) Form ( ) Enhancement ( ) Extension ( ) Online program ( ) Modification ( ) Unknown ( ) Others		
Priority:	( ) high/mandatory ( ) medium/recommended ( ) low/optional		
Data volume:	____________ records	Date to be available:	
Only for Conversion and Interfaces:			
1. Quality of legacy data:	( ) Good ( ) Average ( ) Poor		
2. Complexity of legacy data:	( ) Simple ( ) Average ( ) Complex ( ) Very complex		
3. Complexity of a manual input:	( ) Simple ( ) Average ( ) Complex		
4. Number of Screens:	XX screens		
Only for Interfaces:			
1. Type of interface:	( ) Real-Time ( ) Pseudo-real time ( ) Batch		
2. Direction of interface:	( ) Inbound ( ) Outbound ( ) Both		
3. Frequency:	( ) Daily ( ) Weekly ( ) Monthly ( ) Fortnightly ( ) Other: ____________		
Only for Reports:			
1. The report will be created via:	( ) Report Painter ( ) Report Writer ( ) Query ( ) Program ( ) Info System		
2. Requested characteristics:	( ) Drill-down ( ) Pushbuttons ( ) Sort ( ) Entry files ( ) Others: ________		
Generic question:			
1. Impact of not creating the program:	( ) legal requirements will not be fulfilled ( ) lack of information required for the business ( ) lose functionality compared to the old system ( ) others: ________________		
Development labor estimate:			
Is there any alternative in the standard system:	() Yes () Maybe ( ) No		
Describe the alternative found:			
Reason why this alternative was not acceptable:	( ) Performance problems ( ) Complexity ( ) Others: ____________		
Project cost:		Book cost to:	
Cost accepted by:			
Project management approval on:		Steering committee Approval on:	

3.3 Project Overview Template

Section I: Project Organization	
Organization/Agency:	
Project Manager:	
Project Point of Contact:	
Contact Address:	
Contact Phone:	
Project Mission, Goals and Objectives:	

Section II: Scope		
Business Processes to be addressed:		
Legacy Systems to be retired:		
Check all that apply to the project:	□ ERP Implementation □ RICE Development □ Data Warehousing □ Document Management	□ Business Process Re-Engineering □ Change Management □ End-User Training

Section III: Solution	
ERP/COTS Package:	
Bolt-On Applications:	
Known Interfaces:	
Known Conversions:	
Known Extensions:	

Section IV: System Detail
Complete for each system to be replaced

System:	POC:

Description of system:

Main functions / modules:

Existing interfaces:

Has a decision been made on the replacement system? If so, with which system?

3.4 Technical Design Specifications Template

Section I: Overview				
Area (R/3 Module):			Date (dd/mm/yyyy):	
Requested by:			Phone # / Extension:	
Developer:			Phone # / Extension:	
Program name:				
Title:				
Functional Spec:				
Program type:	( ) Conversion ( ) Batch Interface ( ) Online Interface ( ) Report ( ) Form ( ) Online program ( ) Enhancement ( ) Extension ( ) Standard Modification			
Type:	( ) Creation ( ) Change			
Technique:	( ) Batch-Input ( ) Call Transaction ( ) Direct Input ( ) IDOC ( ) ALE ( ) RFC ( ) CPI-C (X) Others: __________			
Priority:	( ) high-mandatory ( ) medium-recommended ( ) low-optional			
Data volume:	_____________ records		Deadline date:	
Technical description:				

Tables involved:			
Note:			
OBJECT NAME			
Exporting Parameters			
	Field name	*Type*	*Description*
1.			
2.			
3.			
4.			
5.			
6.			
7.			
8.			
9.			
10.			
Importing Parameters			
	Field name	*Type*	*Description*
1.			
Tables Parameters			
	Field name	*Type*	*Description*
1.			
2.			
3.			
4.			
5.			
6.			
7.			
8.			
9.			
10.			
11.			
12.			
13.			
14.			
15.			
16.			
17.			
18.			

Element Assignments			
EXPORT FIELD #	Field name	*Type*	

Element	*Field*	*Source*	*Description*
a.			
b.			
c.			
d.			
e.			
f.			
g.			
h.			
i.			

Element Assignments			
TABLE #	Field name	*Type*	
Element	*Field*	*Source*	*Description*
a.			
b.			
c.			
d.			
e.			

Element Assignments			
TABLE #	Field name	*Type*	
Element			
a.			
b.			
c.			
d.			
e.			
f.			
g.			
h.			
i.			
j.			
k.			
l.			
m.			

Logic Flow:

Workflow Logic

Technical (designer) consultant signature : ______________________________ Date: __/__/____

REVISION HISTORY

Date	**Revision #**	**Description of Change**